DIGITAL FILTER BANK FOLLOWED BY MONOBIT RECEIVERS
FOR SIGNAL PROCESSING

RIGHTS OF THE GOVERNMENT

The invention described herein may be manufactured and used by or for the Government of the United States for all governmental purposes without the payment of any royalty.

BACKGROUND OF THE INVENTION

5   Modern analog-to-digital converters (ADCs) used in radio frequency receiver applications can operate at very high speeds, such as at 3 GHz. However, the fast Fourier transform chip, based on a complex transformation principle, following many known ADCs in digital signal receiver applications, cannot operate at such high speeds due to the complexity of computing the number of sample points examined. The conventional approach to building a wideband digital

10   receiver is to use a polyphase filter to slow down the operation speed of the fast Fourier transform. A specific example to illustrate using a polyphase filter to slow down operation of the fast Fourier transform follows.

The ADC used in this example operates at 3 GHz with 8 bits. A fast Fourier transform is performed on 256 points of digitized data. This fast Fourier transform operation can be written

15   as

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{\frac{-j2\pi kn}{N}} \tag{1}$$

where x(n) is the digitized input data and N = 256. If the output frequency is decimated by 8, k = 0, 8, 16, ... 248 are kept. That is, there are a total of 32 (256/8) outputs. The result from Eq. 1 can be written as

$$X(0) = \sum_{n=0}^{255} x(n)$$

$$X(8) = \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 8n}{256}}$$

$$X(16) = \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 16n}{256}} \tag{2}$$

$$\cdots$$

$$X(248) = \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 248n}{256}}$$

First, arbitrarily consider two frequency components k=16 and k = 248 and rewrite them in slightly different form. The results are

$$
\begin{aligned}
X(16) = \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 16n}{256}} &= \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 2n}{32}} \\
&= [x(0) + x(32) + x(64) + \cdots + x(224)] \\
&+ [x(1) + x(33) + x(65) + \cdots + x(225)]e^{\frac{-j2\pi 2}{32}} \\
&+ [x(2) + x(34) + x(66) + \cdots + x(226)]e^{\frac{-j2\pi 2\times 2}{32}} \\
&+ \cdots \\
&+ [x(31) + x(63) + x(95) + \cdots + x(255)]e^{\frac{-j2\pi 2\times 31}{32}}
\end{aligned}
\tag{3}
$$

and

$$
\begin{aligned}
X(248) = \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 248n}{256}} &= \sum_{n=0}^{255} x(n)e^{\frac{-j2\pi 8n}{32}} \\
&= [x(0) + x(32) + x(64) + \cdots + x(224)] \\
&+ [x(1) + x(33) + x(65) + \cdots + x(225)]e^{\frac{-j2\pi 31}{32}} \\
&+ [x(2) + x(34) + x(66) + \cdots + x(226)]e^{\frac{-j2\pi 31\times 2}{32}} \\
&+ \cdots \\
&+ [x(31) + x(63) + x(95) + \cdots + x(255)]e^{\frac{-j2\pi 31\times 31}{32}}
\end{aligned}
\tag{4}
$$

In the above equations the relation of $e^{-j2\pi n} = 1$ when n = integer is used. Now consider a new quantity, y(n) as

$$y(n) = x(n) + x(n+32) + x(n+64) + \cdots + x(n+224)$$

$$= \sum_{m=0}^{7} x(n+32m) \qquad (5)$$

where n = 0 to 31. The quantity y(n) represents the values in the bracket of Eqs (3) and (4).

Each y value contains a total of 8 data points. This operation is graphically represented in FIG. 1.

In FIG. 1, the 256 input data points shown at 100 are divided into eight 32-point sections, two of

5 which are shown at 101. The beginning data point of each section is shown. These eight

sections are shown stacked at 102 and summed vertically as shown at 103. The results are the 32

y(n) values.

Using these y(n) values, the results from Eq. (2) can be rewritten as

$$X(0) = \sum_{n=0}^{31} y(n)$$

$$X(8) = y(0) + y(1)e^{\frac{-j2\pi}{32}} + y(2)^{\frac{-j2\pi 2}{32}} + \cdots + y(31)e^{\frac{-j2\pi 31}{32}}$$

$$= \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi n}{32}}$$

$$X(16) = y(0) + y(1)e^{\frac{-j2\pi 2}{32}} + y(2)e^{\frac{-j2\pi 2 \times 2}{32}} + \cdots + y(31)e^{\frac{-j2\pi 2 \times 31}{32}} \qquad (6)$$

$$= \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi 2n}{32}}$$

...

$$X(248) = y(0) + y(1)e^{\frac{-j2\pi 31}{32}} + y(2)e^{\frac{-j2\pi 31 \times 2}{32}} + \cdots + y(31)e^{\frac{-j2\pi 31 \times 31}{32}}$$

$$= \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi 31n}{32}}$$

10 All these equations can be written into one as

$$X(8k) = \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi kn}{32}} \qquad (7)$$

where k=0, 1, 2, ..., 31; and n=0, 1, 2, ..., 31.

The output X(8k) can be relabeled as Y(k), thus, Eq. 7 can be written as

$$Y(k) = \sum_{n=0}^{31} y(n)e^{\frac{-j2\pi kn}{32}} \qquad (8)$$

Equation 8 represents a 32-point fast Fourier transform. In order to obtain the outputs of a 256-point fast Fourier transform decimated by 8 on the output, a 32-point fast Fourier transform can achieve the goal. Thus, the design of the fast Fourier transform can be simplified. However, the

5   input signal must be manipulated in order to obtain the desired result.

Further, consider that if one wants to perform an N-point fast Fourier transform and the outputs in the frequency domain are decimated by M, one can achieve the goal by performing a N/M point fast Fourier transform. For this example, a new input format y(n) must be built first. The generalization of the y(n) can be written as

$$y(n) = \sum_{m=0}^{M-1} x(n + mN/M) \qquad (9)$$

where n = 0, 1, 2, ... (N/M)-1. The outputs in the frequency domain can be obtained as

$$X(k) = \sum_{n=0}^{(N/M)-1} y(n)e^{\frac{-j2\pi kn}{N/M}} \qquad (10)$$

This illustrates that when the fast Fourier transform outputs are decimated by M, the outputs can be obtained from N/M point fast Fourier transform. This allows the design of a fast

15   Fourier transform chip to be simplified tremendously.

If the fast Fourier transform uses 256 data points but only every one-eighth of the outputs are kept, there are a total of 32 outputs, and 16 of them carry redundant information. Therefore, only 16 outputs are displayed in FIG. 2. In FIG. 2, the y-axis at 201 represents amplitude in decibels and the x-axis represents frequency. Each filter output is represented by a sinc function,

20   two of which are shown at 200. Only a few side lobes are shown in FIG. 2 and the highest two are only 13 dB below the main lobe, shown at 203. This filter bank has many holes. If an input

signal falls in one of the holes, the receiver will miss it entirely. The shape of this filter is definitely unacceptable.

To widen the individual filters and at the same time suppress the side lobes, a window (or weighting) function can be applied to the input data. There are many different window functions. The one used in the present example is the Parks-McClellan window because it can provide the desired frequency response. The relative amplitude of the window function is shown in FIGs. 3a and 3b. In FIGs. 3a and 3b, the y-axis represents amplitude, in FIG. 3a the x-axis represents time sample at 301 and in FIG. 3b the x-axis at 302 represents frequency. FIG. 3b shows the corresponding frequency domain response with very low pass band ripples at 304 and side lobes lower than 70 dB. This is a desirable filter shape.

The input data x(n) will be modified by the window function h(n). The resulting data $x_m(n)$ used as the input of the fast Fourier transform can be written as

$$x_m(n) = x(n)h(n) \tag{11}$$

where n = 0, 1, 2, ..., 255. As stated previously, the outputs are decimated by 8. Under this condition the modified data can be used in Equation (5) to find the y(n) as

$$y(n) = \sum_{m=0}^{7} x_m(n+32m) = \sum_{m=0}^{7} x(n+32m)h(n+32m) \tag{12}$$

where n = 0, 1, 2, ... (N/M)-1. A few y(n) terms are written as

$$y(0) = x(0)h(0) + x(32)h(32) + \cdots + x(224)h(224)$$
$$y(1) = x(1)h(1) + x(33)h(33) + \cdots + x(225)h(225)$$
$$\cdots \tag{13}$$
$$y(31) = x(31)h(31) + x(63)h(63) + \cdots + x(255)h(255)$$

If a 32-point fast Fourier transform is performed on these y(n) values, 16 individual filters will be generated. Each filter shape is as shown in FIG. 3b.

Now let us consider in more detail the process to generate the y(n) values. Each of these values can be generated from the convolution output of a filter with the input signal. The 256-

5   point window function in the time domain can be written as

$$h(n) = h(255)\delta(n) + h(254)\delta(n-1) + h(253)\delta(n-2) + \cdots + h(0)\delta(n-255) \qquad (14)$$

where the δ function indicates the h(n) value occurs at time n. The impulse sequence of the filter is written in an inverse way. This impulse function can generate the results from Equation (13) through convolution with the input signal. Since the window function shown in FIG. 3a is

10   symmetric in the time domain, this inverse is just a subscript change. The weighting function can be decimated by 32 and the results are 32 individual filters and each filter has eight taps. This filter decimation is often referred to as the polyphase filter. Each of the 32 filters has the response as indicated below.

$$h_0(n) = h(224)\delta(n) + h(192)\delta(n-1) + h(160)\delta(n-2) + \cdots + h(0)\delta(n-7)$$
$$h_1(n) = h(225)\delta(n) + h(193)\delta(n-1) + h(161)\delta(n-2) + \cdots + h(1)\delta(n-7)$$
$$\ldots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (15)$$
$$h_{31}(n) = h(255)\delta(n) + h(223)\delta(n-1) + h(191)\delta(n-2) + \cdots + h(31)\delta(n-7)$$

15   These filters must convolve with the proper input data to obtain the result in Equation (13). In order to obtain the correct data format, the input data must be decimated by 32 also. When the decimated input signal and the decimated filter are convolved and reaching steady state, the output is equal to the result of Equation (13).

The next time the 32-point fast Fourier transform is performed, the input y(n) values to

20   the fast Fourier transform are

$$y(0) = x(32)h(0) + x(64)h(32) + \cdots + x(256)h(224)$$
$$y(1) = x(33)h(1) + x(65)h(33) + \cdots + x(257)h(225)$$
$$\cdots$$
$$y(31) = x(63)h(31) + x(95)h(63) + \cdots + x(287)h(255)$$

(16)

In Eq. 16, the first data point is x(32), thus, the input is shifted 32 points. The hardware to accomplish this goal is shown in FIG. 4. In FIG. 4, there are 32 filters, shown at 400 and each filter has 8 taps. Two cycles of input data are shown at 401 and each cycle contains 32 data

5   points. The outputs at 402 are shown as y(n) and they are used as the input of the fast Fourier transform. The final results in the frequency domain at 403 are represented by Y(k). In this case the inputs are decimated by 32 and the final frequency domain also has 32 outputs. The input data are shifted 32 points, which is also the output frequency bin number. This case is referred to as the critically sampled case. A critically sampled case is that the number of output frequency

10   bin equals to the input data shift. This means that the output sampling rate is 1/M times the input sampling rate where M is the number of input data point shifted.

Each individual filter in FIG. 4 is a finite impulse response (FIR) filter. Filter number 0 with y(0) is shown in FIG. 5. In FIG. 5, the decimated input data points are shown at 500. When the input signal reaches steady state, the output of the filter contains 8 terms. The first and

15   second consecutive filter outputs after steady state are also listed and shown at 501 and 502. The lower line 502 represents the first output, which matches the y(0) output of Equation (13). The upper line 501 represents the second time output where the input data are shifted by 32 points. This result match the y(0) output from Equation (16). The rest of the filter outputs can be obtained in a similar manner.

The shape of the filter bank is shown in FIG. 6. This filter bank has a dynamic range of close to 75 dB, shown at 602, with a bandwidth of about 93.75 MHz, one of which is shown at 603. In these wide band filters, simultaneous signals should also be considered since the probability of two signals falling into one channel is high.

5   To improve the single signal frequency resolution, a phase comparator is used at every output. For example, there are two signals and both are in channel 5 as shown in FIG. 7 at 700, with channel 5 shown at 701. In the graph of FIG. 7, the y-axis at 704 shows the filter response and the x-axis at 705 represents the frequency. Signal 1 is close to channel 4 shown at 702 and signal 2 is close to channel 6 shown at 703. The signals are separated by 1/4 of the channel width and signal 1 is 20 dB stronger than signal 2. The three phase comparators are used to measure the two output frequencies. Phase comparator in channel 4 measures the frequency of signal 1 and phase comparator in channel 6 measures the frequency of signal 2. Channel 5 contains both signals and it is difficult to measure them in one channel. In this design, channels 4 and 6, represented at 702 and 703, respectively, can measure the two frequencies, thus, it is not

15   necessary to obtain the frequency from channel 5. It is desirable to process two signals in one channel by one processor rather than three. The present invention can accomplish this goal.

SUMMARY OF THE INVENTION

The present invention uses narrow-band monobit receivers after a digital filter bank to separate simultaneous signals in one channel. Placing the monobit receiver at the filter outputs

20   improves receiver performance.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a graphical representation of obtaining y values.

FIG. 2 shows outputs from decimated outputs with rectangular window.

FIG. 3a shows a time domain response of a Parks-McClellan window.

FIG. 3b shows a frequency domain response of a Parks-McClellan window.

FIG. 4 shows a channelization approach according to the invention.

FIG. 5 shows an individual polyphase filter according to the invention.

FIG. 6 shows a designed filter bank according to the invention.

FIG. 7 shows two signals in one channel according to the invention.

FIG. 8 shows an arrangement of over sampling by 2 according to the invention.

FIG. 9 shows a modified filter structure according to the invention.

FIG. 10 shows over sampling filters followed by monobit receivers according to the invention.

FIG. 11a shows a four-point kernel function arrangement of the invention.

FIG. 11b shows an eight-point kernel function arrangement of the invention.

FIG. 12 shows a digital filter bank followed by monobit receivers and phase comparators.

DETAILED DESCRIPTION

The present invention uses narrow-band monobit receivers after a digital filter bank to separate simultaneous signals in one channel. The invention improves the capability of wideband digital receivers.

One important factor in building this receiver is the bandwidth of the monobit receiver. In order to process signals falling between channels, the monobit receiver should have a bandwidth wider than 93.75 MHz. One simple approach to increase the bandwidth is to double it to 187.5 MHz, which can be accomplished by doubling the output sampling frequency.

The relation between [1]

$$K = MI \tag{17}$$

where K is the output frequency bins, M is the number of data points shifting per fast Fourier transform operation and I is an integer, which is referred to as the over sampling ratio. In this discussion K = 32, M = 16 and I =2.

When the filter reaches steady state, the input to the fast Fourier transform operator is the

5   same as Equation (13). However, the second cycle of the fast Fourier transform operates, the

input is different from the result of Equation (16), the desired results are

$$y(0) = x(16)h(0) + x(48)h(32) + \cdots + x(240)h(224)$$
$$y(1) = x(17)h(1) + x(49)h(33) + \cdots + x(241)h(225)$$
$$\cdots \hspace{8cm} (18)$$
$$y(31) = x(47)h(31) + x(79)h(63) + \cdots + x(287)h(255)$$

The output starts from x(16) instead of starting from x(32). This discussion is similar to

the discussion in reference 2. FIG. 8 shows that the input at 803 is decimated into 16 outputs,

10   shown at 800, rather than 32. These 16 inputs, represented at 800, are fed into filters 0 to 15,

represented at 801. The inputs to filters 16 to 31 are represented at 802 and are obtained from

delaying the first 16 outputs by one clock cycle. Some of the inputs are listed in this figure. For

example, the input to filter 16 is obtained by delaying input to filter 0 by one clock cycle, thus,

this input is x(16) x(32), ...,represented at 804. Since the input is decimated by 16,

15   comparing with the case decimated by 32, the output speed is doubled.

As shown in FIG. 9, the filters in FIG. 8 are also modified. Between each output,

represented by the line at 800, there are two delay cycles, two of which are shown at 901, rather

than one. The first two consecutive outputs are shown and they match the result of the first

equation in Eq. 13 and Eq. 18. From this arrangement one can see that the output rate to generate

20   Y(k) from the fast Fourier transform is doubled. The output-sampling rate is 187.5 MHz. This is

the input rate to the monobit receiver. Since the outputs from the fast Fourier transform are complex, the monobit receiver has a bandwidth of 187.5 MHz, which is double the 3 dB bandwidth of the output filter.

A simulation demonstrates operational performance of the digital filter followed by

5   monobit receivers arrangement of the present invention. The digital filters are generated through fast Fourier transform operation. The filter shape is the same as shown in FIG. 6. The only difference is the output rate is doubled because each fast Fourier transform is performed by shifting the input 16 points rather than 32 points. However, the sampling rate is not shown in FIG. 6. A threshold should be placed at the outputs of each filter. If the output is lower than the

10   threshold, one should consider that there is no signal output from this channel and its output will not be processed. One possible arrangement is shown in FIG. 10. In FIG.10, the threshold at the outputs of the filters, the filters being shown at 1000, are not shown. Although the 32-point fast Fourier transform generates 32 outputs, shown at 1001, only 16 of them carry independent information, thus only 16 narrow-band monobit receivers are needed, represented at 1002.

15   The channels with signals are processed with the monobit receivers. Only the highest two bits from the real and imaginary parts are used as the input of the monobit receiver. This operation should be equivalent to putting a limiting amplifier at the filter output, although a real limiting amplifier cannot operate on complex signals. This limiting action destroys the amplitude information on the input signals, thus, a threshold is needed at the channel output as

20   discussed in the previous paragraph. In the time domain every 16 outputs from a certain frequency bin are used as one input frame of the monobit receiver. Since the output-sampling rate is 187.5 MHz, the time to collect 16 samples is about 85 ns ($16/187.5 \times 10^6$). This is the same time to collect 256 input data points because $256/3 \times 10^9$ also equal to 85 ns. Thus, both the

digital filter and the monobit receiver process 256 input data points. Since there are only 16

inputs to a monobit receiver the design can be very simple. There are 16 outputs because the

inputs are complex. These 16 outputs cover a bandwidth of 187.5 MHz thus, each individual

output of the monobit receiver is about 11.72 MHz (187.5/16). This implies that the receiver can

5   separate two signals separated by about 12 MHz. However, if the amplitudes of the two signals

from a wide-band digital filter are separated more the 5 dB, the receiver will miss the weak

signal.

Since the fast Fourier transform operation only takes 16 input points, both the Kernel

functions in FIGs. 11a and 11b are used in the simulation. Although two signals in one channel

10   are shown in both approaches, the eight-point Kernel function provides a better result, which

means higher instantaneous dynamic range. The monobit receiver limits the instantaneous

dynamic range, thus, two signals in one channel must have comparable amplitudes to be detected

by the receiver. The instantaneous dynamic range of two signals in different channels is

determined by the shape of the digital filter. In this simulation the threshold is not included. In

15   order to integrate the monobit receiver, a practical configuration and more thorough simulation

are needed.

In the above example, each channel has a width of 11.72 MHz. These bandwidths are

narrow enough to separate two simultaneous signals, but not fine enough to report a frequency

reading. It is desirable to have the capability to report a finer frequency reading. The finer

20   frequency reading can be obtained by using a phase comparator after the monobit receiver. FIG.

12 shows such an arrangement.

Using this approach a single frequency resolution of about 0.5 MHz should be achievable. In order to obtain this resolution the pulse width must increase to a minimum of 512 data points at the input of the receiver, which is about 171 ns.

While the apparatus and method herein described constitute a preferred embodiment of
5   the invention, it is to be understood that the invention is not limited to this precise form of apparatus or method and that changes may be made therein without departing from the scope of the invention, which is defined in the appended claims.